Compact Virtual Keyboard for Abugida Scripts and Efficient Word Completion for Agglutinative Languages

ABSTRACT

Abugida scripts are scripts in which letters are written as a combination of a base grapheme (e.g., consonant) and a modifying grapheme (e.g., vowel sound). Keyboards for abugida languages are complex and often feature flickering (switching) layouts to accommodate the large numbers of letters. Using Tamil as an example abugida script, this disclosure proposes an intuitive, static, keyboard that more efficiently accepts grapheme inputs. Agglutinative languages are languages in which tense, tense-aspects, number, person, conjugation, inflections, prepositions, etc., are expressed by adding suffixes or prefixes to a stem word. Using Tamil as an example agglutinative language, this disclosure proposes more efficient and accurate word-completion techniques.

KEYWORDS

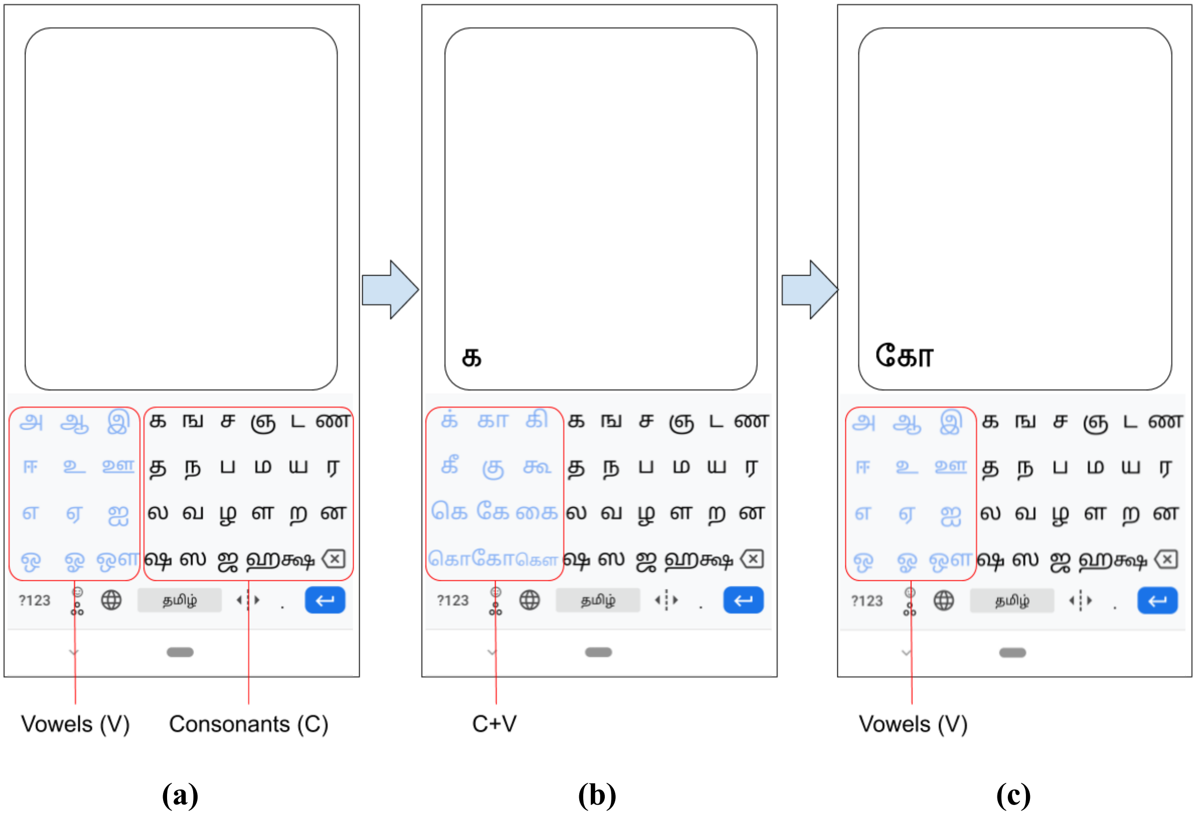
- Virtual keyboard
- Agglutinative languages
- Abugida scripts
- Text entry
- Auto-completion

BACKGROUND

Abugida scripts are scripts in which letters are written as a combination of a base grapheme (e.g., consonant) and a modifying grapheme (e.g., vowel sound). Several languages of India, South-East Asia, Ethiopia, and indigenous northern Canada have abugida scripts. Taking Tamil as an example abugida script, some Tamil letters using the base grapheme க (ka) are as follows.

க்	கா	க (base grapheme)	கெ	கொ
k	kaa	ka	ke	ko

Because every vowel-consonant combination in abugida scripts can be a letter, the number of letters in abugida languages are typically large. For example, Tamil, which has 12 vowels and 18 consonants, has a total (12+1)×(18+1)=247 letters (the number is considerably larger when the letters ஸ ஷ ஜ ஹ ஸ்ரீ க்ஷ, derived from the grantha script and commonly used, are included). It is not atypical for abugida languages to have 30-40 consonants and 10-15 vowels, so that abugida languages with 300-500 letters are hardly unusual. Keyboards for abugida languages are correspondingly complex and often feature flickering (switched) layouts to accommodate the large numbers of letters.



(a) (b) (c)

Fig. 1: Flickering keyboards in Tamil (and other abugida languages)

Fig. 1 illustrates the frequent switching (flickering) of keyboards in Tamil. Fig. 1(a) indicates an initial keyboard layout, comprising vowels (V, in blue) and consonants (C, in black). As shown in Fig. 1(b), upon the entry of a single grapheme (க, ka), the blue section switches to a C+V character-layout representing the variants (க் k; கா kaa; கி ki; கீ kii; …) of the just-entered base grapheme (க, ka). Furthermore, upon selection of a variant (கோ, ko) of the grapheme, the blue section switches back to vowels (Fig. 1(c)). The constant switching (flickering) of the keyboard is strenuous. Another problem with the keyboard shown in Fig. 1(b) is that the C+V section doesn't include the base grapheme; instead, it includes in its place a pure consonant (க், k).

Agglutinative languages are languages in which tense, tense-aspects, number, person, conjugation, inflections, prepositions, etc., are expressed by adding suffixes or prefixes to a stem word. Examples of agglutinative languages are the Dravidian languages (Tamil, Telugu, etc.), Japanese, Korean, Sinhala, Turkic languages (Turkish, Kazakh, Turkmen, Uighur, Kyrgyz, etc.), Finnish, etc.

எழுதுகிறேன்	ezuthu + kiR + En	I write. (simple present)
எழுதிக்கொள்கிறேன்	ezuthu + [i] + koL + kiR + En	I write myself. (reflexive present)
எழுதிக்கொண்டிருக்கிறேன்	ezuthu + [i] + koNdu + iru + kkiR + En	I am writing. (continuous present)
எழுதிவிடுகிறேன்	ezuthu + [i] + vidu + kiR + En	I write [and will finish].

Fig. 2: Phrase variants in an agglutinative language - Tamil

Fig. 2 illustrates example phrase variants in Tamil, an agglutinative language. Various conjugations and inflections of a verb எழுது (to write) are formed by adding appropriate suffixes to the base formation எழுது of the verb. In the example of Fig. 2, the blue suffix indicates the tense; the red suffix indicates the person, number, and gender of the subject; and the brown suffix a tense-aspect. Here, the tense is present, the person first, and the number singular. The tense-aspects of the first three rows are respectively simple, reflexive, and continuous. The aspect of the fourth row does not seem to have an English equivalent; it can be translated as "and will finish" or "will get it done."

It is observed that agglutinative languages use suffixes chained to a base word to convey considerable meaning. In contrast, the non-agglutinative English translation (third column of Fig. 2) uses several words to express a given combination of tense, tense-aspect, number, person, conjugation, preposition, and inflection. Although agglutinative languages have the same notion of a word as in English, the word-possibilities, being a multiplicative combination of tense, person, number, gender, aspect, inflection, etc., are far greater in number.

Auto-completion of words in non-agglutinative languages is typically done using a Markov model. Words are regarded as utterances of a Markov model that emits letters, and the thus-far emitted sequence of letters form a state that determines the probabilities of the utterances of the next letter. For example, the letter "q" in English is followed by "u" with high probability. As another example, the sequence of letters "que" is followed by "u" with high probability, by "s" with slightly less probability, and by "k" with nearly zero probability.

Current techniques for auto-completion, developed mainly for non-agglutinative languages, offer suggestions for the whole word; no suggestions are offered for the next letter or a sub-word stretch of the next few letters, which would be the more appropriate action for agglutinative languages. Auto-completion based on Markov models generally do not account for person-number-gender agreement between subject and verb, resulting in incorrect auto-completion suggestions in agglutinative languages, as shown in Fig. 3 for Tamil.

Incomplete phrase	Auto-completion suggestions	Comments
நாங்கள் பெசி… (we spea...)	நாங்கள் பேசியிருக்கி (we invalid-word)	Invalid word
	நாங்கள் பேசியி...கிறார் (we speak-third-person-plural)	Subject-verb mismatch
	நாங்கள் பேசியி...றேன் (we speak-first-person-singular)	Subject-verb mismatch

Fig. 3: Incorrect auto-completion in Tamil

The first column of Fig. 3 illustrates an incomplete phrase (நாங்கள் பெசி…) input by the user. The second column illustrates auto-completion suggestions made by a popular Tamil-language keyboard. The auto-completion suggestions for the second (incomplete) word, which happens to be a verb, do not account for the person, number, or gender of the first (complete) word, a noun, which results in subject-verb disagreement. The first row illustrates an auto-completion suggestion that is an invalid word in the language. Due to their length, a common feature in agglutinative languages, the auto-completion suggestions are displayed in an elided form using an ellipsis (...), but the part in the middle (the infix) is the most relevant to the part that comes in the end (the suffix). What works well for English auto-completion breaks down for agglutinative languages like Tamil.

DESCRIPTION

For languages with abugida scripts, this disclosure proposes a more intuitive, static keyboard that more efficiently accepts grapheme inputs. For agglutinative languages, this disclosure proposes more efficient and accurate word-completion techniques. The techniques for both are illustrated using Tamil, which is an agglutinative language with an abugida script.

Static keyboard

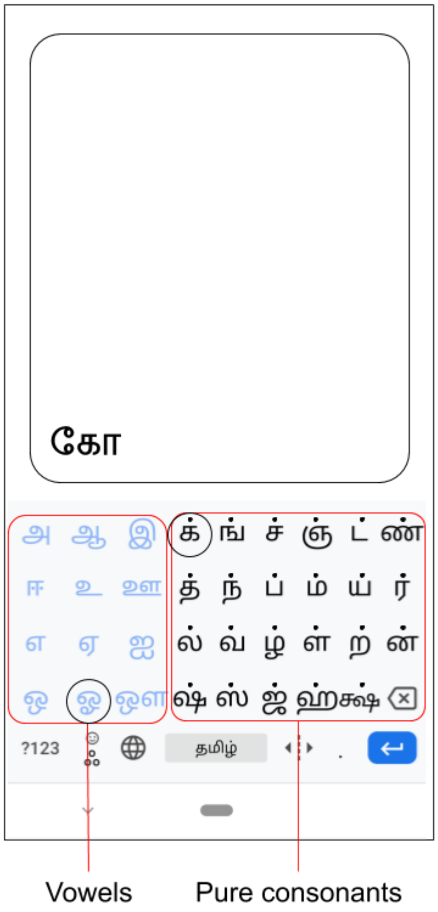


Fig. 4: A static keyboard for Tamil

Fig. 4 illustrates a static keyboard for Tamil, per the techniques of this disclosure. The keyboard is divided into a vowel section and a pure consonant section, both of which are static.

The pure consonant section comprises consonants with the dot (புள்ளி, *pulli*), which indicates pronunciation devoid of vowel sound (க் k; ங் ng; ச் ch; …). A vowel-consonant grapheme is constructed by touching the corresponding vowel and pure consonant buttons sequentially or simultaneously. For example, the displayed character கோ (ko) is constructed by touching the circled buttons. Effectively, the character கோ is constructed using the equation

கோ = க் + ஓ,

such that letters are constructed using their phonemes rather than their graphemes. In this respect, the proposed static keyboard is similar to the Korean Hangul keyboard.

க் (k) variants	ங் (ng) variants	…	க்ஷ் (ksh) variants
க =க் + அ (ka)	ங = ங் + அ (nga)		க்ஷ = க்ஷ் + அ (ksha)
கா = க் + ஆ (kaa)	ஙா = ங் + ஆ (ngaa)		க்ஷா = க்ஷ் + ஆ (kshaa)
கி = க் + இ (ki)	ஙி = ங் + இ (ngi)		க்ஷி = க்ஷ் + இ (kshi)
கீ = க் + ஈ (kii)	ஙீ = ங் + ஈ (ngii)		க்ஷீ = க்ஷ் + ஈ (kshii)
…	…		…
கௌ = க் + ஔ (kau)	ஙௌ = ங் + ஔ (ngau)		க்ஷௌ = க்ஷ் + ஔ (kshau)

Fig. 5: All the letters of Tamil can be constructed with the static keyboard

All the letters of Tamil can be generated by the described static keyboard using the table of Fig. 5, where the binary plus (+) operator indicates sequential or simultaneous key-presses of its two operand-letters.

Accurate and efficient Auto-completion

Per the techniques, auto-completion suggestions for a given word agree in person, number, and gender with completed words thus far entered. Further, to avoid the appearance of the ellipsis in the middle of auto-completion suggestions, suffixes (infixes) are suggested one at a time. It is only upon the selection of a suffix that the next suffix in the chain is suggested.

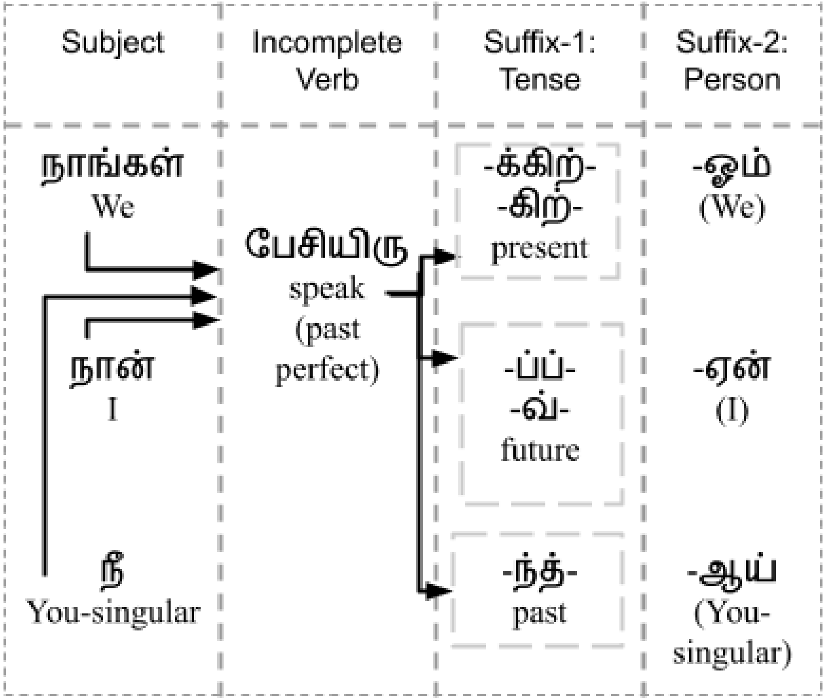


Fig. 6: Accurate and efficient auto-completion

Fig. 6 illustrates accurate and efficient auto-completion, per the techniques of this disclosure. The user enters a subject-noun (an entry from column one, e.g., நாங்கள், we; நான், I; ந°, you-singular) and an incomplete verb (பெசியிரு, speak, column 2). The valid pathways to completing the agglutinative verb are indicated by the arrows. Auto-completion suggestions are morphemes (suffixes or infixes) rather than letters. Suffixes or infixes are presented one-by-one for selection by the user. For example, suffix-1, tense (column 2) is presented, and only after the selection of tense is suffix-2, person (column 3) presented. In this manner, the earlier-mentioned ambiguity of ellipsis is avoided. Unlike previous techniques, pathways that do not conform to the verb conjugation rules of the language or to the morphology of the language are absent from the list of auto-completion suggestions.

CONCLUSION

Using Tamil as an example abugida script, this disclosure proposes an intuitive, static, virtual keyboard that more efficiently accepts grapheme inputs. Using Tamil as an example agglutinative language, this disclosure proposes more efficient and accurate word-completion techniques.

REFERENCES

[1] https://www.learntamil.com/ accessed on Dec. 15, 2020.